

日常生活における行動の自動識別に関する一考察

A study on automatic identification of behaviors in daily life

中嶋 瑛¹⁾
指導教員 林 誠治¹⁾

1) 拓殖大学工学部 機械・電子システム工学専攻 林研究室

キーワード：機械学習，行動認識，加速度センサー，NNC

1. はじめに

日々の生活行動においてどのような行動がどれくらい行われているのかを知ることは、自らの健康状態を評価する一助となり、更なる健康への意識改革をもたらす要因となり得る。またこれを介護施設などに利用すれば、利用者の行動別運動量を測定することが可能になる。

本研究では、人間の行動により得られる加速度データを入力とし、RNN および CNN を用いた機械学習による行動の自動識別を行ったので報告する。

2. Neural Network Console について

NNC は Sony が開発したディープラーニング・ツールであり、従来のツールと大きく違う点はコーディングをなしにディープラーニングを実装することができる点にある。使用方法やツールの導入手法などは Sony が公式で説明しているため、環境構築の際の手間は掛かりにくい。本ディープラーニング・ツールの主な特徴として、以下の5つあげられる。

- ドラック&ドロップの簡単編集によるネットワーク構造の作成と変更
- ニューラルネットワーク構造の自動探索
- 学習の進捗状況がリアルタイムで確認可能
- 学習したニューラルネットワークを履歴として一覧で閲覧可能
- クラウド版も実装されており、Google chrome への接続で、どの OS からでも NNC が使用可能

NNC でのデータセットに関しては、NNC が定める CSV フォーマットのデータセット形式に対応している。データセットの基本構成としては、ヘッダを表す1行目とデータを表す2行目以降で構成される。ヘッダの各セルは、CSV ファイルの各列のデータの変数名、次元インデックス、ラベル名を示す。

3. 本研究での行動認識における機械学習

3.1 使用する加速度データについて

使用する加速度データは、Android Studio で測定用に使えるように本研究で作成および調整したアプリを使用して収集する。測定に使用したスマートフォン端末は Zenfon AR であり、内蔵センサーは ICM20602 Accelerometer を使用している。

3.2 加速度取得アプリ作成

アプリはスマートフォンの加速度センサーのサンプリング周波数を調整するプログラム [1] を参考にし、Android Studio を用いて本研究用にレイアウトやデータをカスタマイズしたものである。本アプリの動作フローチャートを図1に示す。

3.3 加速度データの収集

本研究では基礎研究のこともあり、行動の種類として“歩く”、“走る”、“止まる(静止)”の3つの動作を選んだ。人体へのスマートフォンの装着の様子を図2に示す。取付場所は腰回りの左手前とし、ゴムバンドとスマートフォンポーチで固定した。また、毎回の装着時には、上下左右の位置がそれぞれ3cm以内、傾きが15度以内となるようにした [2]。取り付けたスマー

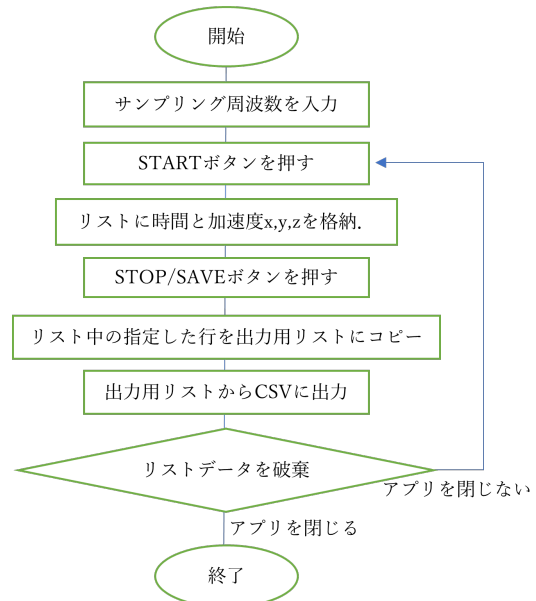


図1 スマホアプリの動作フローチャート

トフォンのセンサー3軸を図3に示す。



図2 スマートフォンの装着位置

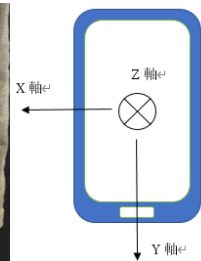


図3 3軸の向き

3軸加速度データの収集は50Hzの一定速度とし、一つの行動動作を約20秒間に設定した。3つの行動パターンに対して、それぞれ5回計測した。測定場所は屋外のアスファルト路面上を直線的に移動して計測した。

3.4 収集した加速度のCSVデータについて

実際にアプリで収集した加速度データの例を図4に示す。左から収集時刻、x軸、y軸、z軸の加速度である。また、図5は機械学習に入力する学習/評価用データの例であり、図4のデータに対して、各x,y,z加速度データを[G] ($1G = 9.80665 m/s^2$)の単位に変更したものである [3]。図5の実際のデータ長(フレーム長)を2.56秒に設定し、各フレーム間は50%のオーバーラップを行うことにより、学習用データとして488フレーム、評価用データとして210フレームを得た。

	A	B	C	D
1	15.42.42.850	-0.26373	17.57439	3.33508
2	15.42.42.869	-0.67314	13.87534	3.63675
3	15.42.42.889	-1.09332	9.070175	2.59647
4	15.42.42.909	-0.56659	5.708725	1.15515
5	15.42.42.929	0.16124	4.017212	0.39619
6	15.42.42.949	1.402634	6.156433	-0.2383
7	15.42.42.969	1.692337	9.205444	-0.8811
8	15.42.42.989	1.452911	11.81992	-1.4378
9	15.42.43.009	1.278137	11.18546	-0.8643
10	15.42.43.029	0.152863	8.974411	0.24536
11	15.42.43.049	-0.04106	7.772522	0.07059
12	15.42.43.069	0.668808	7.060242	-0.7422

図4 修正前のデータ例

	A	B	C
1	-0.02689	1.792089	0.340084
2	-0.06864	1.414891	0.370845
3	-0.11149	0.9249	0.264786
4	-0.05778	0.582128	0.117793
5	0.016442	0.409642	0.040401
6	0.143029	0.627781	-0.0243
7	0.17257	0.938694	-0.08985
8	0.148156	1.205296	-0.14661
9	0.130334	1.140599	-0.08814
10	0.015588	0.915135	0.02502
11	-0.00419	0.792577	0.007198
12	0.068199	0.719944	-0.07589

図5 修正後のデータ例

	y_0	y_1	y_2	Recall
ylabel=0	42	0	0	1
ylabel=1	0	84	0	1
ylabel=2	0	0	84	1
Precision	1	1	1	
F-Measures	1	1	1	

	y_0	y_1	y_2	Recall
ylabel=0	42	0	0	1
ylabel=1	0	84	0	1
ylabel=2	0	0	84	1
Precision	1	1	1	
F-Measures	1	1	1	

図10 正解率の高いRNNとCNNの学習モデルの評価結果

3.5 NNCを使用した機械学習

3.5.1 RNNのネットワーク

本研究で扱う行動データは3軸の加速度データで時系列データであるため、時間関係の特徴を抽出することができるRNNを利用する。実測した学習データを用いて、図6に示すLSTMのネットワーク構成により機械学習を行う。

3.5.2 CNNのネットワーク

CNNは画像や動画のデータを対象にすることが多いが、行動識別においても相性が良く識別精度も高いとされている。本研究では、128行3列となっている実測データを3行128列の形にNNCで転置した後、図7に示す1次元の畳み込みネットワーク(1D CNN)を基本構成とし、NNCの自動探査を用いて機械学習を行う。

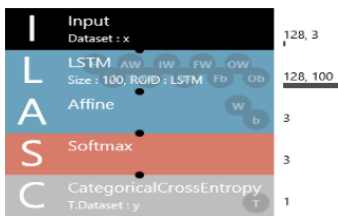


図6 LSTMを使用したネットワークの基本構成

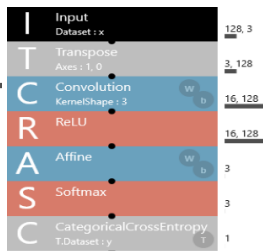


図7 1DCNNの基本構成

4. 実行結果と考察

4.1 正解率が高いRNNとCNNのネットワーク

図8のRNNのネットワークでは、基本構成にMaxPooling, BatchNormalizationが追加されたことで、VALIDATIONエラーが減少し、正解率は1.0が得られた。図9のCNNのネットワークでは、基本構成の段階で正解率が1.0となったが、Convolutionの直前にMaxPoolingが追加されることにより、正解率は1.0のまま学習コストが大幅に削減された。

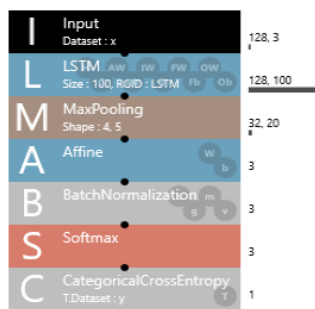


図8 正解率の高いRNNの学習モデル

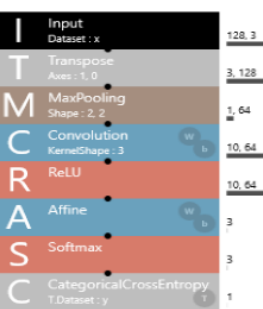


図9 正解率の高いCNNの学習モデル

4.2 RNNとCNNの比較結果

“歩く”, “走る”, “止まる(静止)”の3つの動作に対して、図10の混合行列の結果からわかるように、正解率はRNN, CNN共に1.0が得られ、再現率および適合率においても1.0という高い精度結果が得られた。

一方、図11と図12を比較すると、RNNの方がEPOCH数に対してエラーの減少が早いことより、CNNより低いEPOCH数で高い精度に到達できる。しかしながら、図12のCostMultiplyAddはCNNの方が少なくRNNの約1/1373となる。実際にEPOCH数をRNNは20, CNNは100として比較しても学習の時間はRNNで47秒, CNNで42秒とCNNの方が速くなった。

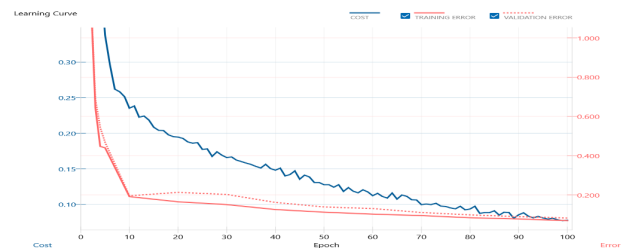


図11 正解率が高いRNNの学習評価グラフ

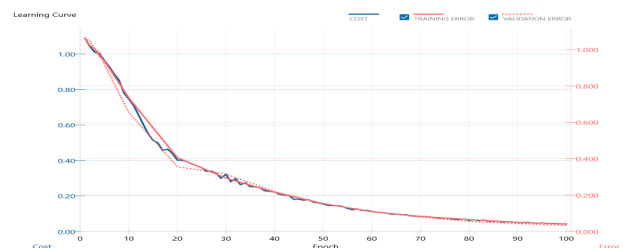


図12 正解率の高いCNNの学習評価グラフ

5. まとめ

本研究では、3つの行動を分類する行動識別をRNNとCNNを用いて行った結果、RNNとCNNとともに高い認識精度を得ることができ、適切な学習ネットワークの構成がある程度見出せたといえる。

今後の課題としては、1) 行動分類数の増加, 2) 様々な状況での収集, 3) 被験者数の増加などが考えられ、本研究で得たネットワーク構成をもとにモデルをさらに学習することで、日常生活における自動識別の完成を目指したい。

参考文献

- [1] “加速度センサのサンプリング周波数を調整できるAndroidアプリを作った話”, <https://qiita.com/miwazawa/items/6635488b2ace8db889a2> (参照:2020-11-23)
- [2] “HASC2010corpus”, <http://hasc.jp/hc2010/HASC2010corpus/hasc2010corpus.html> (参照:2020-12-13)
- [3] 河口信夫ら, “HASC Challenge2010:人間行動理解のための装着型加速度センサコーパスの構築”, DICOMO2011シンポジウム, 平成23年7月
- [4] “Neural Network Consoleドキュメント”, <https://support.dl.sony.com/docs-ja/> (参照:2020-12-15)

統計情報	RNN	CNN
Output	13,834	2,119
CostParameter	43,535	1,963
CostAdd	51,209	646
CostMultiply	3	0
CostMultiplyAdd	5,275,520	3,840
CostDivision	3	3
CostExp	3	3
CostIf	12,800	896

図13 RNNとCNNの学習コスト